

# CODE-VERIFICATION TECHNIQUES FOR PARTICLE-IN-CELL SIMULATIONS WITH DIRECT SIMULATION MONTE CARLO COLLISIONS

Brian A. Freno  
William J. McDoniel  
Christopher H. Moore  
Neil R. Matula  
Sandia National Laboratories

17th World Congress on Computational Mechanics  
10th European Congress on Computational Methods in Applied Sciences and Engineering  
July 19–24, 2026

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.

# Outline

- Introduction
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
- Numerical Examples
- Summary

# Outline

- Introduction
  - Collisional Plasma Dynamics
  - Code Verification
  - This Work
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
- Numerical Examples
- Summary

# Collisional Plasma Dynamics

- Important for many scientific and engineering applications
  - **Hypersonic & reentry air plasmas** affecting heat loads and radiation
  - **Plasma devices** such as lightning arresters and plasma switches
  - **Pulsed power** for simulating Sandia flagship experimental facilities
  - **Semiconductor & thin-film plasmas** for etching and deposition
- Modeled via particle-in-cell (PIC) with collision algorithm (MCC/DSMC)
  - Solve Maxwell's equations to compute electromagnetic field on grid
  - Solve particle equations of motion due to Lorentz force and collisions
  - Interpolate field onto particles, distribute particle properties onto grid
  - Model particle collisions with direct simulation Monte Carlo (DSMC)

# Verification and Validation

Credibility of computational physics codes requires verification and validation

- **Validation** assesses how well models represent physical phenomena
  - Compare computational results with experimental results
  - Assess suitability of models, model error, and bounds of validity
- **Verification** assesses accuracy of numerical solutions against expectations
  - *Solution verification* estimates numerical error for particular solution
  - *Code verification* assesses correctness of numerical-method implementation

Code verification is the focus of this work

# Importance of Code Verification

## Consequences of incorrectly implemented numerical methods:

- Wasted computational expense:
  - Numerical errors **decrease slower** with mesh refinement than they should
  - Numerical errors **do not decrease** with mesh refinement
- Numerical solutions **differ significantly** from exact solutions
- **Application failure** from incorrect numerical results
- Consequences may not be obvious

## Benefits of code verification:

- Assess suitability and correctness of numerical methods
- Compare accuracy and efficiency of different algorithms
- Set expectations and estimate error for solution verification

# Code Verification

Code verification assesses correctness of numerical-method implementation

- Continuous equations are numerically discretized

$$\mathbf{r}(\mathbf{u}) = \mathbf{0} \quad \rightarrow \quad \mathbf{r}_h(\mathbf{u}_h) = \mathbf{0}$$

- Discretization error is introduced in solution

$$\mathbf{e} = \mathbf{u}_h - \mathbf{u}$$

- Discretization error should decrease as discretization is refined

$$\lim_{h \rightarrow 0} \mathbf{e} = \mathbf{0}$$

- More rigorously, should decrease at an expected rate

$$\|\mathbf{e}\| \approx Ch^p$$

- Measuring error requires exact solution – usually unavailable

# Manufactured Solutions: Overview

Manufactured solutions are popular alternative

- Manufacture an arbitrary solution  $\mathbf{u}_{MS}$
- Insert manufactured solution into continuous equations to get residual term

$$\mathbf{r}(\mathbf{u}_{MS}) \neq \mathbf{0}$$

- Add residual term to discretized equations

$$\mathbf{r}_h(\mathbf{u}_h) = \mathbf{r}(\mathbf{u}_{MS})$$

to coerce solution to manufactured solution

$$\mathbf{u}_h \rightarrow \mathbf{u}_{MS}$$

# Manufactured Solutions: An Example

Consider Laplace's equation in 1D  $r(u) = \frac{\partial^2 u}{\partial x^2} = 0$

Discretized by finite differences

$$r_h(u_h) = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} = 0$$

Manufacture the arbitrary solution  $u_{\text{MS}}(x) = \sin(\pi x)$ , such that

$$r(u_{\text{MS}}) = \frac{\partial^2 u_{\text{MS}}}{\partial x^2} = -\pi^2 \sin(\pi x)$$

Solve  $r_h(u_h) = r(u_{\text{MS}})$ , such that

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} = -\pi^2 \sin(\pi x_i)$$

The discretization error is  $e = u_h - u_{\text{MS}}$

# Code Verification

- Code verification assesses correctness of numerical-method implementation
- Most rigorously measures rate at which error decreases with refinement
- Error requires exact solution – usually unavailable
- Manufactured solutions are popular alternative:  $\mathbf{r}(\mathbf{u}) = \mathbf{0} \rightarrow \mathbf{r}(\mathbf{u}) = \mathbf{r}(\mathbf{u}^M)$ 
  - Manufacture an arbitrary solution  $\mathbf{u}^M$
  - Insert manufactured solution into equations to get residual term  $\mathbf{r}(\mathbf{u}^M)$
  - Add residual term to equations to make manufactured solution a solution
- For *collisionless* PIC simulations, few instances of code verification exist
- Significant code-verification challenges with *collisional* PIC simulations:
  - Discretization errors from space and time discretization
  - Statistical sampling error from finite number of computational particles
  - Stochasticity from collision modeling – considering random subset of collisions

# Existing Work: Collisionless Particle-in-Cell Simulations

- Riva et al., *Physics of Plasmas* (2017), 10.1063/1.4977917
  - Modify particle weights to achieve manufactured distribution function
    - Particles move independently of manufactured distribution function
    - Particle weights modified according to Vlasov equation
  - 1D, electrons
  - Measure electric field error
  - Multiple approaches with varying expense to measure dist. function error
  - Many runs per discretization
- Tranquilli et al., *Journal of Computational Physics* (2022), 10.1016/j.jcp.2021.110751
  - Extend the approach of Riva et al. to 2D, electrons and ions
  - Measure charge density, electric field, and electric potential errors
  - Derive expected convergence rates for statistical sampling errors in fields
  - Argue against the need to measure error in distribution function
  - Single run per discretization

# This Work: Collisional Particle-in-Cell Simulations

- Apply method of manufactured solutions to equations of motion
  - Weights are unmodified – no negative weights, collision-algorithm modifications
  - Obtain manufactured particle positions and velocities at each time step
    - Inversely query cumulative manufactured distribution function
- Balance collision-algorithm velocity change with manufactured source term
  - Average outcomes from multiple collision-algorithm runs at each time step
  - Compute analytic expected change in velocity due to collisions
    - Manufacture cross section and anisotropy
- Apply method of manufactured solutions to Poisson equation
  - Manufacture electric scalar potential
- Compute field errors and particle errors
  - Single run per discretization
- Demonstrate approach for collisional and collisionless PIC simulations

# Outline

- Introduction
- Governing Equations
  - Particle-in-Cell Method Overview
  - Equations of Motion for Charged Particles
  - Maxwell's Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
- Numerical Examples
- Summary

# Particle-in-Cell Method Overview

- Place (weighted) computational particles randomly in phase space (according to distribution function)
- Interpolate particle charge onto spatial mesh
- Solve Maxwell's equations on spatial mesh for electromagnetic field
- Interpolate electromagnetic field onto particles
- For each particle, integrate equations of motion due to
  - Lorentz force from electromagnetic field
  - Collisions between particles

# Equations of Motion for Charged Particles (single species, electrostatic)

Equations of motion for each particle  $p$ :

$$\dot{w}_p(t) = 0, \quad \dot{\mathbf{x}}_p(t) = \mathbf{v}_p(t), \quad \dot{\mathbf{v}}_p(t) = \frac{\mathbf{F}_p(t)}{m} + \left( \frac{\Delta \mathbf{v}_p(t)}{\Delta t} \right)_{\text{coll}}$$

- $w_p$ ,  $\mathbf{x}_p$ , and  $\mathbf{v}_p$  are computational particle weight, position, and velocity
- $\mathbf{F}_p(t) = q\mathbf{E}_p(t)$  is electrostatic Lorentz force,  $\mathbf{E}_p(t) = \mathbf{E}(\mathbf{x}_p(t), t)$
- $\mathbf{E}$  is electric field
- $m$  and  $q$  are species mass and charge
- $(\Delta \mathbf{v}_p / \Delta t)_{\text{coll}}$  is instantaneous change in  $\mathbf{v}_p$  per  $\Delta t$  due to collision algorithm

Distribution function evolution approximates Boltzmann equation:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \frac{\mathbf{F}}{m} \cdot \frac{\partial f}{\partial \mathbf{v}} = \left( \frac{\partial f}{\partial t} \right)_{\text{coll}}$$

- $f(\mathbf{x}_p, \mathbf{v}_p, t)$  is particle distribution function,  $(\partial f / \partial t)_{\text{coll}}$  is collision term

# Maxwell's Equations (electrostatic case – negligible magnetic field)

Gauss's law

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}$$

→

$$\Delta\phi = -\frac{\rho}{\epsilon_0}$$

Gauss's law for magnetism

$$\nabla \cdot \mathbf{B} = 0$$

↑

Faraday's law of induction

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}_x}{\partial t}$$

→

$$\mathbf{E} = -\nabla\phi$$

Ampère's circuital law

$$\nabla \times \mathbf{B} = \mu_0 \left( \mathbf{J} + \epsilon_0 \frac{\partial \mathbf{E}}{\partial t} \right)$$

- Charge conservation  $\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{J} = 0$
- Charge density  $\rho(\mathbf{x}, t) = q \int_{-\infty}^{\infty} f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}$
- Electric current density  $\mathbf{J}(\mathbf{x}, t) = q \int_{-\infty}^{\infty} f(\mathbf{x}, \mathbf{v}, t) \mathbf{v} d\mathbf{v}$
- $\epsilon_0$  and  $\mu_0$  are permittivity and permeability of free space

# Outline

- Introduction
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
  - Manufactured Particle Distribution Function
  - Manufactured Solutions through the Equations of Motion
  - Manufactured Source Term for Binary Elastic Collisions
  - Manufactured Solutions for the Poisson Equation
- Error Analysis
- Numerical Examples
- Summary

# Manufactured Particle Distribution Function

Assume  $f^M$  takes the form of 3D analog of previous work:

$$f^M(\mathbf{x}, \mathbf{v}, t) = f_{\mathbf{x}}(\mathbf{x}, t) f_{\mathbf{v}}(\mathbf{v}, t),$$

where

$$f_{\mathbf{v}}(\mathbf{v}, t) = \prod_{i=1}^3 f_{v_i}(v_i, t), \quad f_{v_i}(v_i, t) = \frac{2}{\sqrt{\pi}} \frac{v_i^2}{\hat{v}_i(t)^3} e^{-v_i^2/\hat{v}_i(t)^2}, \quad \int_{-\infty}^{\infty} f_{v_i}(v_i, t) dv_i = 1,$$

and

$$f_{\mathbf{x}}(\mathbf{x}, t) = N \prod_{i=1}^3 f_{x_i}(x_i, t), \quad \int_0^{L_{x_i}} f_{x_i}(x_i, t) dx_i = 1, \quad \int_V f_{\mathbf{x}}(\mathbf{x}, t) d\mathbf{x} = N$$

- $N$  is the number of physical particles in the volume  $V = \prod_{i=1}^3 L_{x_i}$
- Separability of  $f_{\mathbf{x}}(\mathbf{x}, t)$  imposed for convenience
- $f_{\mathbf{v}}(\mathbf{v}, t)$  is deliberately non-Maxwellian,  $\hat{v}_i(t)$  incorporates time variation

# Manufactured Solutions through the Equations of Motion

Apply **method of manufactured solutions** to equations of motion:

$$\dot{\mathbf{x}}_p = \mathbf{v}_p + \dot{\mathbf{x}}_p^M - \mathbf{v}_p^M, \quad \dot{\mathbf{v}}_p = \frac{q}{m} \mathbf{E}_p + \left( \frac{\Delta \mathbf{v}_p}{\Delta t} \right)_{\text{coll}} + \dot{\mathbf{v}}_p^M - \frac{q}{m} \mathbf{E}_p^M - \left( \frac{\Delta \mathbf{v}_p^M}{\Delta t} \right)_{\text{coll}}$$

- **Weights unmodified** – no negative weights, collision-algorithm modifications
- At  $t = 0$ , for component  $i$ , take uniform random samples  $\xi_{x_{i_p}}, \xi_{v_{i_p}} \in [0, 1]$
- Inversely query cumulative dist. functions to obtain  $x_{i_p}^M(t)$  and  $v_{i_p}^M(t)$ :

$$F_{x_i}(x_{i_p}^M(t), t) = \xi_{x_{i_p}}, \quad F_{v_i}(v_{i_p}^M(t), t) = \xi_{v_{i_p}}$$

- Differentiate to obtain  $\dot{x}_{i_p}^M(t)$  and  $\dot{v}_{i_p}^M(t)$
- In general,  $\dot{\mathbf{x}}_p^M \neq \mathbf{v}_p^M$
- $\mathbf{E}^M = -\nabla \phi^M$  is electric field from manufactured potential  $\phi^M$
- $(\Delta \mathbf{v}_p^M / \Delta t)_{\text{coll}}$  is manufactured collision term

# Manufactured Source Term for Collisions

## Velocity equation

$$\dot{\mathbf{v}}_p = \frac{q}{m} (\mathbf{E}_p - \mathbf{E}_p^M) + \frac{(\Delta \mathbf{v}_p)_{\text{coll}} - (\Delta \mathbf{v}_p^M)_{\text{coll}}}{\Delta t} + \dot{\mathbf{v}}_p^M$$

Requires velocity changes due to collisions at each time step:

- $(\Delta \mathbf{v}_p)_{\text{coll}}$  due to **stochastic** collision algorithm
- $(\Delta \mathbf{v}_p^M)_{\text{coll}}$  due to corresponding **deterministic** manufactured source term

At each time step, make collision-algorithm outcome **less stochastic**:

- Run collision algorithm  $N_{\text{avg}}$  independent times, average velocity change
- Replace  $(\Delta \mathbf{v}_p)_{\text{coll}}$  with  $\langle \Delta \mathbf{v}_p \rangle_{\text{coll}} = \frac{1}{N_{\text{avg}}} \sum_{k=1}^{N_{\text{avg}}} (\Delta \mathbf{v}_p)_{\text{coll}}^k$
- Derive expected velocity change for each particle:  $(\Delta \mathbf{v}_p^M)_{\text{coll}} = \langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}$

## Expected Change in Velocity (binary elastic collisions)

Post-collision velocities are obtained from momentum and energy conservation:

$$\mathbf{v}'_p = \frac{1}{2}(\mathbf{v}_q + \mathbf{v}_p + g\mathbf{n}), \quad \mathbf{v}'_q = \frac{1}{2}(\mathbf{v}_q + \mathbf{v}_p - g\mathbf{n}), \quad \mathbf{n} = \begin{Bmatrix} \cos \epsilon \sin \chi \\ \sin \epsilon \sin \chi \\ \cos \chi \end{Bmatrix},$$

where  $g = |\mathbf{v}_p - \mathbf{v}_q| = |\mathbf{v}'_p - \mathbf{v}'_q|$  is relative speed

The velocity change for particle  $p$  is  $\Delta\mathbf{v}_p = \mathbf{v}'_p - \mathbf{v}_p = \frac{1}{2}(\mathbf{v}_q - \mathbf{v}_p + g\mathbf{n})$

Compute expected velocity change across possible collision partners:

$$\langle \Delta\mathbf{v}_p^M \rangle_{\text{coll}} = \frac{\frac{N_p^{\text{cell}} - 1}{2} \int_{\Delta V} \int_{-\infty}^{\infty} \int_0^{2\pi} \int_0^{\pi} P_{\text{coll}}(g)(\mathbf{v}_q - \mathbf{v}_p + g\mathbf{n}) f^M(\mathbf{x}_q, \mathbf{v}_q, t) p(\chi, \epsilon) d\chi d\epsilon d\mathbf{v}_q d\mathbf{x}_q}{\int_{\Delta V} \int_{-\infty}^{\infty} \int_0^{2\pi} \int_0^{\pi} f^M(\mathbf{x}_q, \mathbf{v}_q, t) p(\chi, \epsilon) d\chi d\epsilon d\mathbf{v}_q d\mathbf{x}_q}$$

$P_{\text{coll}}(g) = \frac{\sigma(g)gw\Delta t}{\Delta V}$  is collision likelihood,  $p(\chi, \epsilon)$  is probability density function,  $\sigma(g)$  is cross section

$\langle \Delta\mathbf{v}_p^M \rangle_{\text{coll}}$  is deterministic, should be computed analytically – complicated by  $g$

→ Manufacture anisotropy and cross section

# Manufactured Anisotropy (to avoid dependency on $g$ due to $\mathbf{np}$ )

Model probability density function as separable:  $p(\chi, \epsilon) = p_\chi(\chi)p_\epsilon(\epsilon)$ , where

$$\int_0^{2\pi} \int_0^\pi p(\chi, \epsilon) d\chi d\epsilon = 1, \quad \int_0^{2\pi} p_\epsilon(\epsilon) d\epsilon = 1, \quad \int_0^\pi p_\chi(\chi) d\chi = 1$$

Azimuthally symmetric scattering:  $p_\epsilon(\epsilon) = \frac{1}{2\pi}$

In expression for  $\langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}$ ,

$$g \int_0^{2\pi} \int_0^\pi \mathbf{np}(\chi, \epsilon) d\chi d\epsilon = \frac{g}{2\pi} \int_0^{2\pi} \int_0^\pi \mathbf{np}_\chi(\chi) d\chi d\epsilon = g \left\{ 0, 0, \underbrace{\int_0^\pi p_\chi(\chi) \cos \chi d\chi}_{=0} \right\}$$

Avoid dependency on  $g$  from anisotropy:  $\int_0^\pi p_\chi(\chi) \cos \chi d\chi = 0$

Avoid isotropy:  $p_\chi(\chi) \neq \frac{\sin \chi}{2}$

For  $F_{p_\chi}^{-1}$ , use ansatz  $p_\chi(\chi) = (C_0 + C_1 \cos \chi + C_2 \cos^2 \chi + C_3 \cos^3 \chi) \sin \chi$

$C_2$  and  $C_3$  satisfy constraint,  $C_0$  and  $C_1$  minimize  $\int_0^\pi (p_\chi(\chi) - \bar{p}_\chi(\chi))^2 d\chi$

# Manufactured Cross Section (to evaluate $\langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}$ analytically)

With  $\int_0^\pi p_\chi(\chi) \cos \chi d\chi = 0$ ,

$$\langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}} = \frac{w\Delta t(N_p^{\text{cell}} - 1)}{2\Delta V} \int_{-\infty}^{\infty} \sigma(g) g(\mathbf{v}_q - \mathbf{v}_p) f_{\mathbf{v}}(\mathbf{v}_q, t) d\mathbf{v}_q$$

If  $\sigma(g) = \sum_{n=0}^{N_\sigma-1} \sigma_n g^{2n-1}$ ,

$$\langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}} = \frac{w\Delta t(N_p^{\text{cell}} - 1)}{2\Delta V} \sum_{n=0}^{N_\sigma-1} \sigma_n \mathbf{f}_n,$$

where  $\mathbf{f}_n = \int_{-\infty}^{\infty} g^{2n}(\mathbf{v}_q - \mathbf{v}_p) f_{\mathbf{v}}(\mathbf{v}_q, t) d\mathbf{v}_q$  can be computed analytically

Numerical and manufactured source terms balanced with manufactured  $p_\chi$  and  $\sigma$

# Additional Error Metrics for Collisions

- Coding errors are **not detected** if the expected collision error is zero
- A necessary (but insufficient) condition for zero expected collision error is

$$\int_0^{2\pi} \int_0^\pi \mathbf{n} p(\chi, \epsilon) d\chi d\epsilon = \mathbf{0}$$

- Additionally measure convergence of  $p_\chi(\chi)$  and  $p_\epsilon(\epsilon)$  by recording

$$\chi = \cos^{-1} \frac{g'_z}{g}, \quad \epsilon = \tan^{-1} \frac{g'_y}{g'_x},$$

and measuring difference between analytic and empirical CDFs

- Additional metrics detect problems with
  - Scattering angles
  - Constant scaling of the variance of the collision error

# Manufactured Solutions for the Poisson Equation

Manufacture the electric scalar potential  $\phi^M(\mathbf{x}, t)$  and solve

$$\Delta\phi = -\frac{\rho}{\epsilon_0} + \Delta\phi^M + \frac{\rho^M}{\epsilon_0},$$

where  $\Delta\phi^M$  is evaluated analytically

Evaluate manufactured charge density analytically as well:

$$\rho^M(\mathbf{x}, t) = q \int_{-\infty}^{\infty} f^M(\mathbf{x}, \mathbf{v}, t) d\mathbf{v} = q f_{\mathbf{x}}(\mathbf{x}, t)$$

# Outline

- Introduction
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
  - Discretization Parameters
  - Error Sources
- Numerical Examples
- Summary

## Discretization Parameters

- Discretize spatial domain with uniform cells of length  $\Delta x_i$
- Numerically integrate with time-step size  $\Delta t$
- Represent physical particles with  $N_p$  computational particles
- Run the collision algorithm  $N_{\text{avg}}$  times, average the outcomes

Refine these quantities together:

$$\Delta x_i \sim \Delta t \sim h,$$

$$N_p \sim h^{-s}, \quad N_{\text{avg}} \sim h^{-r}, \quad N_{\text{cell}} \sim h^{-3}, \quad N_p^{\text{cell}} \sim h^{-(s-3)}$$

Measure discrete  $L^1$ ,  $L^2$ , and  $L^\infty$  norms, refine so error in  $L^\infty$  is at most  $\mathcal{O}(h^2)$

- Collisional:  $s = 7 \rightarrow N_p \sim h^{-7}, \quad r = 5 \rightarrow N_{\text{avg}} \sim h^{-5}$
- Collisionless:  $s = 5 \rightarrow N_p \sim h^{-5}$

# Error Sources

## Field Quantities

- Trilinear basis functions:  $\mathcal{O}(h^2)$  for  $\phi$ ,  $\mathcal{O}(h^2)$  recovery for  $\mathbf{E} = -\nabla\phi$
- Sampling error:  $\mathcal{O}(N_p^{-1/2})$  for  $\phi$ ,  $\mathcal{O}(N_p^{-1/2}h^{-1/2})$  for  $\mathbf{E}$  (Tranquilli et al., 2022)
- Particle-position error:  $\mathcal{O}(h^{p_x})$  for  $\phi$  and  $\mathbf{E}$

## Particles (second-order-accurate velocity-Verlet time integration)

$$\mathbf{v}_p^{n+1/2} = \mathbf{v}_p^n + \frac{1}{2} \left( \Delta t \frac{q}{m} (\mathbf{E}_p - \mathbf{E}_p^M)^n + \langle \Delta \mathbf{v}_p \rangle_{\text{coll}}^n - \langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}^n + \Delta t (\dot{\mathbf{v}}_p^M)^n \right) + \boldsymbol{\tau}_{\mathbf{v}_p}^n,$$

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t (\mathbf{v}_p + \dot{\mathbf{x}}_p^M - \mathbf{v}_p^M)^{n+1/2} + \boldsymbol{\tau}_{\mathbf{x}_p}^n,$$

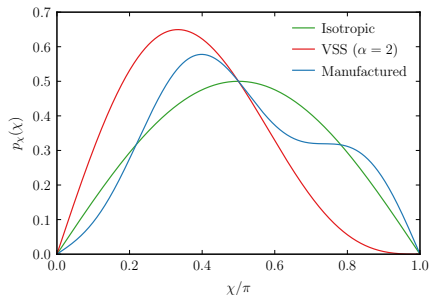
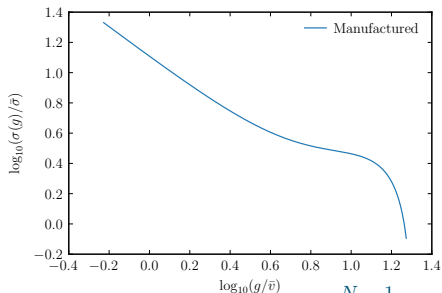
$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^{n+1/2} + \frac{1}{2} \left( \Delta t \frac{q}{m} (\mathbf{E}_p - \mathbf{E}_p^M)^{n+1} + \langle \Delta \mathbf{v}_p \rangle_{\text{coll}}^{n+1/2} - \langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}^{n+1/2} + \Delta t (\dot{\mathbf{v}}_p^M)^{n+1} \right) + \boldsymbol{\tau}_{\mathbf{v}_p}^{n+1/2}$$

- Per-step collision error:  $\mathbf{e}_{\text{coll}_p}^n = \langle \Delta \mathbf{v}_p \rangle_{\text{coll}}^n - \langle \Delta \mathbf{v}_p^M \rangle_{\text{coll}}^n$
- Per-step Lorentz-force acceleration error:  $\mathbf{e}_{\text{acc}_p}^n = \Delta t \frac{q}{m} (\mathbf{E}_p - \mathbf{E}_p^M)^n$
- Time-integration truncation errors:  $\boldsymbol{\tau}_{\mathbf{v}_p}^n + \boldsymbol{\tau}_{\mathbf{v}_p}^{n+1/2} \sim \mathcal{O}(\Delta t^3)$ ,  $\boldsymbol{\tau}_{\mathbf{x}_p}^n \sim \mathcal{O}(\Delta t^3)$

# Outline

- Introduction
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
- Numerical Examples
  - Overview
  - Collisional Particle-in-Cell Simulations
  - Collisionless Particle-in-Cell Simulations
  - Coding Errors
- Summary

# Manufactured Cross Section and Anisotropy



- **Cross section**  $\sigma(g) = \sum_{n=0}^{N_\sigma-1} \sigma_n g^{2n-1}$ 
  - $N_\sigma = 3$ ,  $\bar{\sigma} = 1 \text{ \AA}^2$ ,  $\bar{v} = 10^6 \text{ m/s}$

- **Anisotropy**  $F_{p_\chi}^{-1}, p_\chi(\chi) = (C_0 + C_1 \cos \chi + C_2 \cos^2 \chi + C_3 \cos^3 \chi) \sin \chi$ 
  - $\bar{p}_\chi(\chi) = \alpha \cos(\chi/2)^{2\alpha-1} \sin(\chi/2)$ ,  $1 \leq \alpha \leq 2$ , (variable soft sphere)
  - **Isotropic** ( $\alpha = 1$ ), **anisotropic** ( $\alpha > 1$ )

# Manufactured Solutions and Discretizations

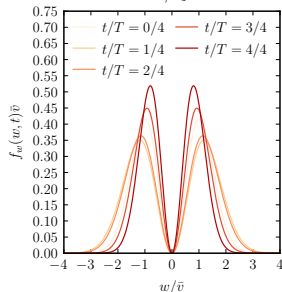
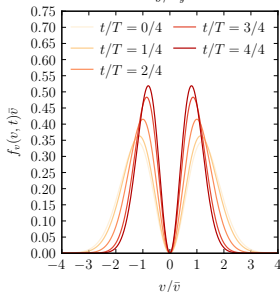
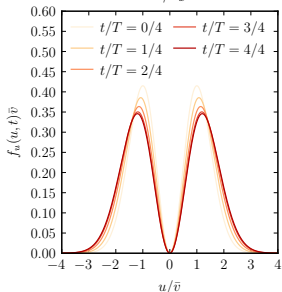
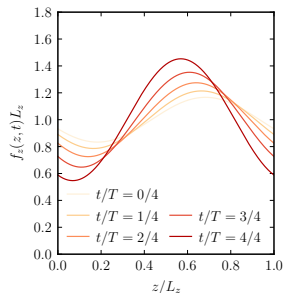
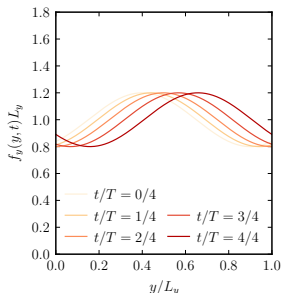
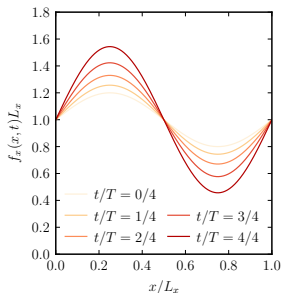
Particle distribution function:  $f^M(\mathbf{x}, \mathbf{v}, t) = f_{\mathbf{x}}(\mathbf{x}, t)f_{\mathbf{v}}(\mathbf{v}, t)$ ,

$$f_{\mathbf{x}}(\mathbf{x}, t) = N \prod_{i=1}^3 f_{x_i}(x_i, t), \quad f_{\mathbf{v}}(\mathbf{v}, t) = \prod_{i=1}^3 f_{v_i}(v_i, t),$$

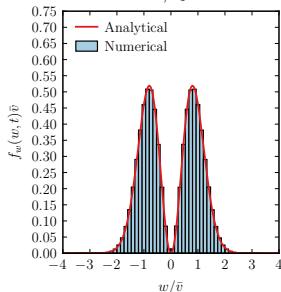
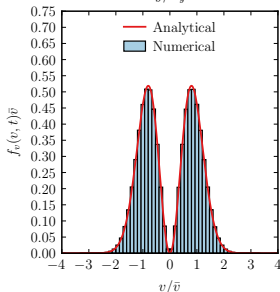
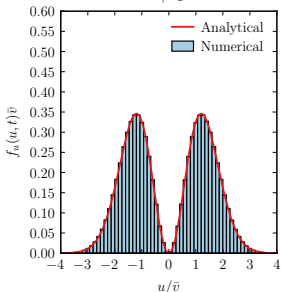
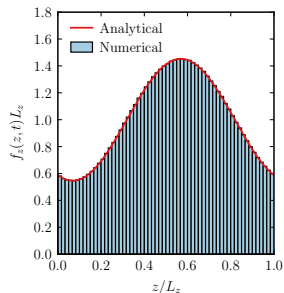
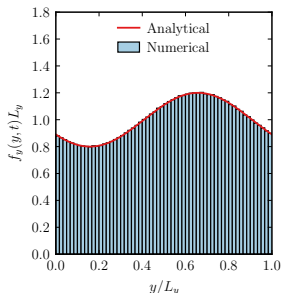
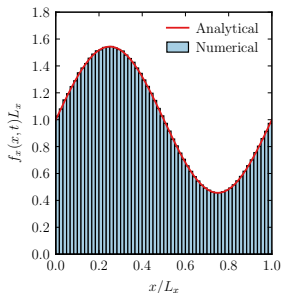
Potential:  $\phi^M(\mathbf{x}, t) = \bar{\phi}e^{t/(2T)} \sin\left(2\pi\left[\frac{x}{L_x} - \frac{1}{7}\right]\right) \sin\left(2\pi\left[\frac{y}{L_y} - \frac{1}{5}\right]\right) \sin\left(2\pi\left[\frac{z}{L_z} - \frac{1}{3}\right]\right)$

$$\begin{aligned} \bar{v} &= 10^6 \text{ m/s}, & L_{x_i} &= 3/2 \text{ m}, & T &= L_{x_i}/(10\bar{v}), & N &= 10^{20} \text{ particles} \\ q &= e, & m &= 3 \times 10^8 m_e, & \bar{\phi} &= 10^{10} \text{ V}, & & \text{periodic BCs} \end{aligned}$$

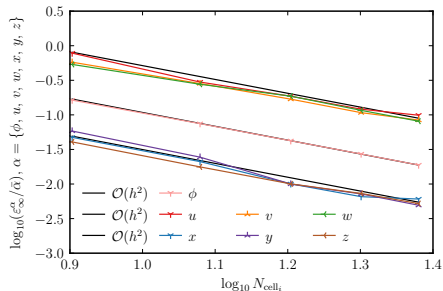
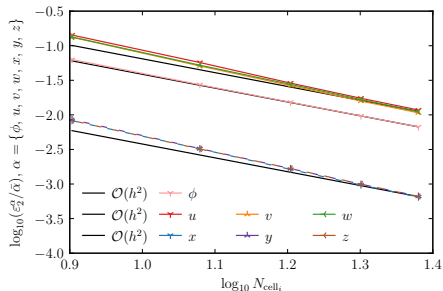
Disc.	$T/\Delta t$	$N_{\text{cell}_i}$	$N_{\text{cell}}$	Collisional			Collisionless	
				$N_{\text{avg}}$	$N_p$	$N_p/N_{\text{cell}}$	$N_p$	$N_p/N_{\text{cell}}$
1	8	8	512	32	10,240	20.00	10,240	20.00
2	12	12	1,728	243	174,960	101.25	77,760	45.00
3	16	16	4,096	1,024	1,310,720	320.00	327,680	80.00
4	20	20	8,000	3,125	6,250,000	781.25	1,000,000	125.00
5	24	24	13,824	7,776	22,394,880	1620.00	2,488,320	180.00

Particle Distribution Function  $f_M(\mathbf{x}, \mathbf{v}, t) = f_x(\mathbf{x}, t)f_v(\mathbf{v}, t)$ 

# Particle Distribution Function $f_M(\mathbf{x}, \mathbf{v}, t) = f_x(\mathbf{x}, t)f_v(\mathbf{v}, t), t = T$

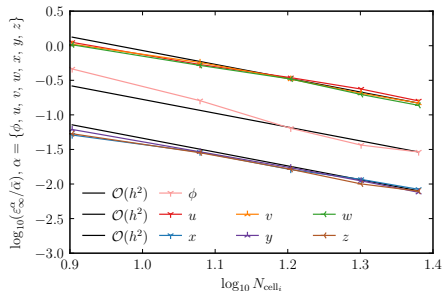
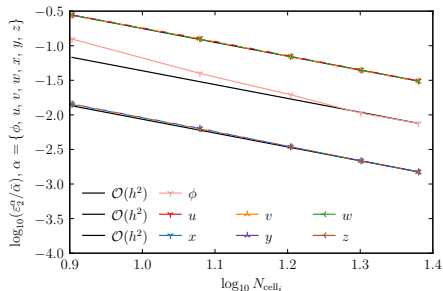


# Error Convergence at $t = T$ : Collisional, Uncoupled



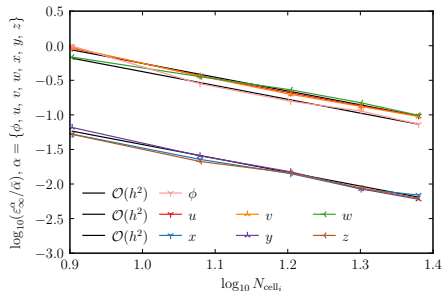
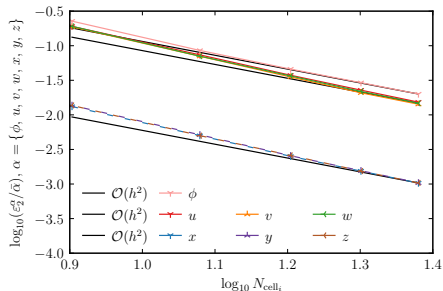
- Particles and field **uncoupled** (single run)
  - Field does not affect particles ( $q/m = 0$ )
  - Particles do not affect field ( $q = 0$ )
- Particle error due to **collisions**, **time integration**
  - Velocity and position:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$
- Field error due to **basis functions**
  - Potential:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$

# Error Convergence at $t = T$ : Collisional, One-Way Coupled



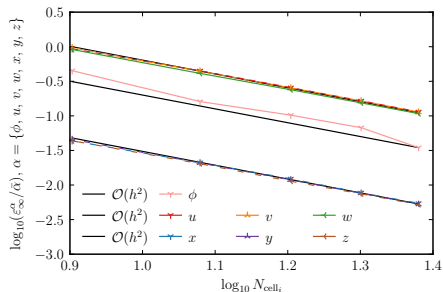
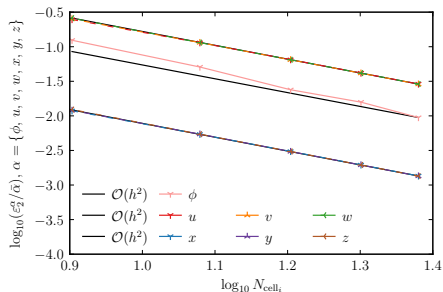
- Particles and field **one-way coupled** (2 separate runs)
  - Field affects particles but is not affected by particles ( $q/m \neq 0, q = 0$ )
  - Particles affect field but are not affected by field ( $q \neq 0, q/m = 0$ )
- Particle error due to **collisions**, **time integration**, **field basis function error**
  - Velocity and position:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$
- Field error due to **basis functions**, **finite sampling**, **collisions**, **integration**
  - Potential:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$

# Error Convergence at $t = T$ : Collisional, Fully Coupled



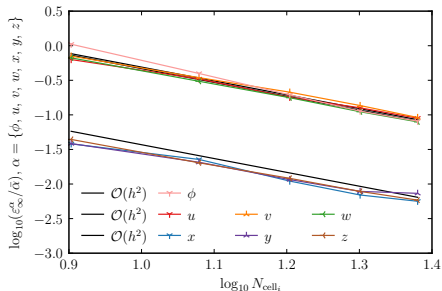
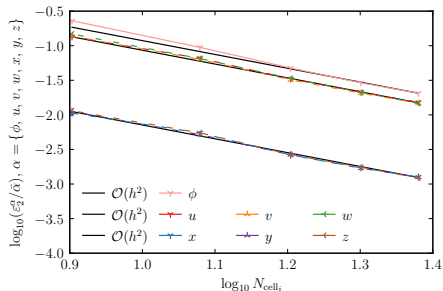
- Particles and field **fully coupled** (single run)
  - Field affects particles ( $q/m \neq 0$ )
  - Particles affect field ( $q \neq 0$ )
- Particle error due to **collisions**, **time integration**, **all field errors**
  - Velocity and position:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$
- Field error due to **basis functions**, **finite sampling**, **all particle errors**
  - Potential:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$

# Error Convergence at $t = T$ : Collisionless, One-Way Coupled



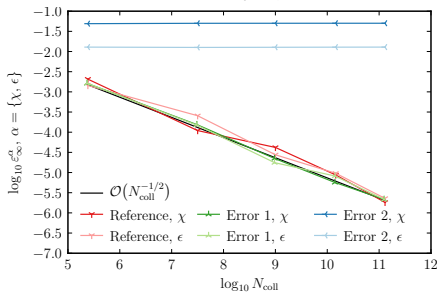
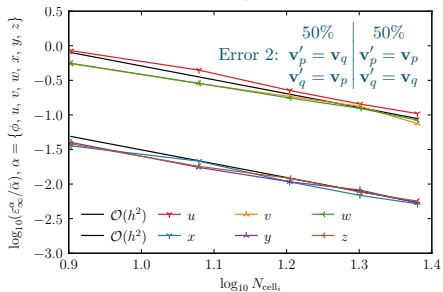
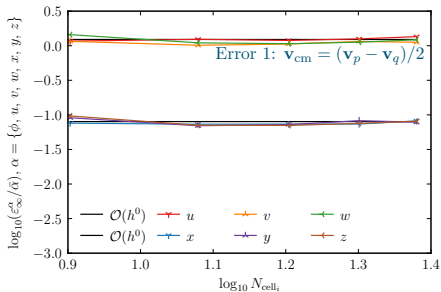
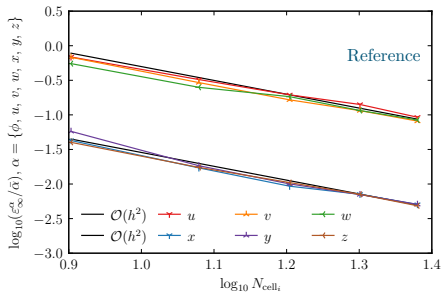
- Particles and field **one-way coupled** (2 separate runs)
  - Field affects particles but is not affected by particles ( $q/m \neq 0, q = 0$ )
  - Particles affect field but are not affected by field ( $q \neq 0, q/m = 0$ )
- Particle error due to **time integration**, **field basis function error**
  - Velocity and position:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$
- Field error due to **basis functions**, **finite sampling**, **particle time integration**
  - Potential:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$

# Error Convergence at $t = T$ : Collisionless, Fully Coupled



- Particles and field **fully coupled** (single run)
  - Field affects particles ( $q/m \neq 0$ )
  - Particles affect field ( $q \neq 0$ )
- Particle error due to **time integration**, **all field errors**
  - Velocity and position:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$
- Field error due to **basis functions**, **finite sampling**, **all particle errors**
  - Potential:  $\mathcal{O}(h^2)$  in  $L^1, L^2, L^\infty$

# Coding Errors (Collisions Only)



# Outline

- Introduction
- Governing Equations
- Manufactured Solutions for PIC Simulations with DSMC
- Error Analysis
- Numerical Examples
- **Summary**
  - Closing Remarks

# Closing Remarks

- Presented code-verification approach for collisional PIC simulations
- Added manufactured source terms to equations of motion (weights unmodified)
- Manufactured distribution function, potential, cross section, and anisotropy
- Computed manufactured source terms analytically, averaged collisions
- Achieved expected convergence rates with one simulation per discretization
  - For collisional and collisionless cases with varying degrees of coupling
  - With and without coding errors

## Acknowledgments

- Duncan McGregor
- David Sirajuddin
- Tom Smith

## References

- B. Freno, W. McDoniel, C. Moore, N. Matula  
Code-verification techniques for particle-in-cell simulations with DSMC collisions  
*Journal of Computational Physics* (2026) [arXiv:2602.15130](https://arxiv.org/abs/2602.15130)
- F. Riva and C. Beadle and P. Ricci  
A methodology for the rigorous verification of particle-in-cell simulations  
*Physics of Plasmas* (2017)
- P. Tranquilli and L. Ricketson and L. Chacón  
A deterministic verification strategy for electrostatic particle-in-cell algorithms in arbitrary spatial dimensions using the method of manufactured solutions  
*Journal of Computational Physics* (2022)

